Programmierbarer humanoider Roboter

Rob-230

Firmware: SV2x0x

Vorwort ... 3
Wichtige Hinweise zum Betrieb des Rob-230 Systems ... 4
Lieferumfang ... 5
Anschlüsse, Bedien- ,Sensor- und Signalelemente ... 5
Erstmalige Inbetriebnahme ... 6
Vorbereitung für die Erstellung eines Steuerprogramms ... 6
Konfiguration des Terminalprogramms „ShamCom“ ... 6
Ausführ-, Programmier- und Firmware-Update-Modus ... 7
Programmzeileneditor ... 8
Funktionsweise des Programmzeileneditors ... 8
Editor- und Ausführbereich ... 9
Zusammenfassung der Editorbefehle ... 9
System Utility Menü ... 10
Utility Menü: Sprachausgabemodul ... 11
Utility Menü: Spracherkennungsmodul ... 11
Programmiersprache für die Steuerprogrammerstellung ... 12
Grundlegender Aufbau einer Programmzeile ... 12
Beispiele für mögliche Steuerprogrammzeilen ... 12
Anwender- und Systemvariable ... 13
Steuerprogrammbefehle ... 16
Befehle für zeitabhängige Geh- und Drehbewegungen ... 16
Befehle für zeitabhängige Geh- und Drehbewegungen mit vorzeitiger Abbruchmöglichkeit ... 17
Befehle für die Bewegung der Arme und des Oberkörpers ... 17
Befehle für die Ausführung einer kompletten Bewegungssequenz mit Soundausgabe ... 18
Befehle für die Registrierung und Auswertung von Sensorsignalen ... 19
Befehle für das Setzen von Parametern ... 19
Allgemeine Befehle der Programmsteuerung ... 20
Transferbefehle ... 20
Arithmetische Befehle ... 20
Befehle um Werte zu vergleichen ... 21
Befehle für die Ein- und Ausgabe über die serielle Schnittstelle ... 21
Befehle für die Ansteuerung des Sprachausgabemoduls ... 22
Befehl für die Ansteuerung des Spracherkennungsmoduls ... 22
Sonstige Befehle ... 22
Firmware-Update durchführen ... 23
Programmbeispiele ... 24
Einfacher autonomer Spaziergang mit Hinderniserkennung ... 24
Auswertung der Umgebungsgeräusche mit entsprechender Reaktion ... 24
Annäherung eines Objektes auswerten mit Reaktion bei Abstandunterschreitung ... 24
Mögliche Störungen ... 25
Roboter reagiert nicht nach dem Einschaltvorgang ... 25
Kontakt ... 25

Vorwort

Der Rob-230 ist in seiner Art das bisher einzige in dieser Preisklasse erhältliche humanoide Robotersystem, welches mit zahlreichen Sensoren, einer Spracherkennungs- und Sprachausgabefunktion, sowie einer leicht erlernbaren Programmerstellungssoftware ausgestattet ist.
Das Hauptziel, das wir bei der Entwicklung von Rob-230 verfolgt haben ist die Möglichkeit einer einfachen Auswertung aller Sensorsignale, sowie komplexe Bewegungsabläufe mit nur einem Befehl abzurufen.
Die Programmerstellungssoftware (Editor, Interpreter) welche komplett im Robotersystem integriert wurde, ähnelt etwas einem Basic-Dialekt und ist selbst für jeden Programmiereinsteiger in kürzester Zeit erlernbar.
Durch die Implementierung von speziellen aber sehr effektiven Steuer- und Sensorauswertbefehlen, können bereits mit wenigen Programmzeilen sehr beeindruckende autonome Steuerabläufe des Roboters erstellt werden.

Es stehen folgende Sensoren und Module für die Verwendung in Ihrem Steuerprogramm zur Verfügung:

- Entfernungsmessung (Ultraschallsensor) im Bereich von 3 bis über 300cm mit einer Auflösung von 1cm.
- Geräuschmessung (Mikrofon) und Auswertung der Geräuschlautstärke.
- Bewegungsmelder (2 Stück IR-Passiv-Sensoren, seitlich rechts und seitlich links) für die Ortung von Personen. Durch die automatische Auswertung dieser Signale, kann auf sehr einfache Weise eine Aussage über die Bewegungsintensität, sowie einer von 3 möglichen Richtungen des Standortes einer Person gemacht werden.
- Helligkeitsmessung (2 Stück Lichtsensoren, seitlich rechts und seitlich links). Auswertung der Leuchtstärke in 1024 Stufen, sowie dessen Richtung.
- Betriebsspannungshöhe (Spannungsmessung der Akkuzellen mit einer Auflösung von ca. 15mV)
- Fernbedienung (Auswertung der von der IR-Fernbedienung gesendeten Signale)
- Spracherkennungseinheit mit integrierter Lernfunktion.
- Synthetische englische Sprachausgabe „Text to Speech“.

Für ein schnelles Erfolgserlebnis bei der Steuerprogrammerstellung durch den Anwender ist es immer vorteilhaft die Zeiten für die Programmerstellung im Editor bis zur Programmausführung so kurz wie möglich zu halten.
Dies erreichen wir durch die Verwendung eines im Robotersystem integrierten Editors und Interpreters, der das eingegebene Steuerprogramm sofort nach dessen Speicherung ausführen kann.
Für die Steuerprogrammerstellung wird neben einem PC mit USB oder RS232 Schnittstelle, nur noch ein Terminalprogramm benötigt, welches VT100 kompatibel sein sollte. Das Terminalprogramm „ShamCom“ ist bereits im Lieferumfang enthalten und läuft auf allen Windows-Systemen.

Terminalprogramme sind für alle populären Betriebssystem wie Windows, Dos, Linux, Apple usw. erhältlich, und werden auch für zukünftige Betriebssysteme erhältlich sein. Dadurch ist gewährleistet, dass Ihr Rob-230 System auch in Zukunft immer verwendbar und einsatzfähig bleibt.

Die vom Rob-230 System zur Verfügung gestellten Resourcen (Sensoren, mechanische Ansteuerung) sind so vielfältig, dass sich einerseits unzählige programmierbare Möglichkeiten für den Anwender ergeben, wie andererseits die Möglichkeit besteht die Firmware stetig mit neuen Funktionen zu erweitern.
Aus diesem Grund haben wir uns dazu entschlossen einen „Bootloader“ in das Rob-230 System zu integrieren, was ein Update der neuesten Firmware ermöglicht.

Wichtige Hinweise zum Betrieb des Rob-230 Systems

- Unbedingt NUR den org. mitgelieferten Ladenetzadapter zum Laden der Akkus und für den Programmiermodus verwenden. Das System ist exakt auf die zur Verfügung gestellte Lade- und Betriebsspannung eingestellt. Da aus technischen Gründen auf ein Verpolungsschutz verzichtet werden musste, führt ein versehentliches Vertauschen der Spannungsrichtung eines fremden Ladenetzteiles unweigerlich zur Zerstörung der internen Elektronik.

- Die maximale Ladedauer bei entladenem Akku (beide LEDs auf der Frontseite blinken schnell) liegt bei 15 Stunden. ACHTUNG: Das Überschreiten der Ladezeiten kann die Akkukapazität drastisch verringern!

- Die Gelenke des Robotersystems sollten möglichst NICHT von Hand bewegt werden, auch nicht im ausgeschalteten Zustand. Bewegen Sie diese dennoch mit der Hand und übergehen dabei einen merklichen Widerstand, so kann dies zu einem Getriebeschaden führen.

- Sie sollten das System während des Betriebs ständig beobachten und mögliche Gefahrenquellen wie abschüssige Untergründe vermeiden. Steckt Ihr Rob-230 mal in einem Hindernis fest, so befreien Sie ihn möglichst schnell aus seiner misslichen Lage, um unnötige Belastungen für das Getriebe zu vermeiden. Auch in diesem Zusammenhang sollte der Betrieb nur auf möglichst glattem Untergrund wie Parkett, Laminat oder Teppichböden mit niedrigem Flor erfolgen.

- Sie haben die Möglichkeit das Robotersystem stetig mit der neuesten Firmware zu aktualisieren. Achten Sie dabei, dass Sie dies NUR mit dem auf der Frontseite eingesteckten Ladenetzteil durchführen. Ein vorzeitiges Unterbrechen der Stromversorgung, während die Firmware aktualisiert wird, kann den internen Flash-Speicher so durcheinander bringen, dass eine komplette Neuprogrammierung des Bootloaders vorgenommen werden muss.

- Bei der Spracherkennung ist zu beachten, dass diese naturgemäß am Besten funktioniert je geringer der Abstand bei einem Zuruf zum Roboter ist und dass mögliche Störgeräusche vermieden werden. Sind alle Voraussetzungen für eine vorteilhafte Spracherkennung wie z.B. ein identischer Klang, und eine der Entfernung angepasste Lautstärke erfüllt, so sind auch Entfernungen für Zurufe von über 3m möglich.

- Bei einem notwendigen Austausch der Sicherung (2A flink) verwenden Sie immer eine entsprechende Ersatzsicherung mit genau den gleichen Kenndaten.

- Bitte beachten bei der Entsorgung von Elektronikprodukten wie dem Rob-230 und dessen Akkuzellen auf die gesetzlich vorgeschriebenen Bestimmungen. Alle Komponenten müssen fachgerecht entsorgt werden und gehören nicht in den Hausmüll.

- Das Robotersystem wurde versiegelt, womit jeglicher Eingriff nicht mehr von einer Gewährleistungspflicht abgedeckt wird.

Lieferumfang

- Rob-230 inkl. bereits eingesetzten Akkuzellen (1800mAh).
- Ladeadapter, welcher auch für den Programmiermodus die notwendige Betriebsspannung liefert.
- Infrarot Fernbedienung.
- Firmware-Update Aktivierungsstecker (5 poliger DIN Stecker).
- Serielles Verbindungskabel mit 9-poligem SUB-Stecker (RS232 kompatibel).
- USB to SERIAL Converter, für den Betrieb an einer USB-Schnittstelle.
- CD-Rom auf der dieser Bedienungsanleitung und ein universelles Terminalprogramm für Windows Betriebssysteme.

Anschlüsse, Bedien- ,Sensor- und Signalelemente

Ladestromanschluss
Auf der Rückseite links von Rob-230 befindet sich eine Ladestrombuchse womit Sie die Akkus bei entladenem Zustand mit maximal 15 Stunden aufladen können. Während des Ladevorganges ist die Ausführung eines Steuerprogrammes nicht möglich.

Betriebsstromschalter
Auf der rechten Seite befindet sich der Betriebsstromschalter (Kippschalter) um ein zuvor gespeichertes Steuerprogramm auszuführen (Schalterstellung oben = Programmausführung aktiviert).

Stromanschluss für den Betrieb im Programmiermodus
Auf der Vorderseite befindet sich ebenfalls eine Strombuchse. Bei eingestecktem Ladeadapter wird nach ca. 4 sec der Programmiermodus gestartet, womit Sie in Verbindung mit einem betriebsbereiten Terminalprogram, z.B. ein zuvor gespeichertes Steuerprogramm weiter bearbeiten können. Für eine Verbindung mit Ihrem PC besitzt das Rob-230 System eine RS232 Schnittstelle, die als 3 polige Klinkenbuchse unterhalb der Strombuchse ausgeführt ist. Ein notwendiges im Lieferumfang beigelegtes Adapterkabel führt die Signale auf einen genormten 9-poligen SUB-Stecker. Besitzt Ihre Rechner jedoch keine RS232 Schnittstelle mehr, so verwenden Sie bitte den beigelegten USB to SERIAL Converter, den Sie nur noch installieren und dazwischenschalten müssen.

RS232 Schnittstelle
Die gleich unter der Stromanschluss befindliche RS232 Schnittselle ist als 3 polige Klinkenbuchse ausgeführt, und stellt die Verbindung zu einem Terminalprogramm her.

IR-Passiv-Sensoren (Bewegungsmelder) mit integrierten Lichtsensoren
Auf der Vorderseite befinden sich zwei IR-Passiv-Sensoren, die Bewegungen von Personen und deren Position registrieren können. Zusätzlich enthält jeder Sensor intern noch einen lichtempfindlichen Widerstand (LDR) für die Erfassung der Umgebungshelligkeit aus zwei leicht versetzten Richtungen.

Signal LEDs
Direkt oberhalb der IR-Passiv-Sensoren befindet sich jeweils eine rote LED, die unterschiedliche Zustände im Ausführ- und Firmware-Update-Modus signalisieren.
Während der Steuerprogrammausführung signalisieren diese die Richtung einer Person, die durch die IR-Passiv-Sensoren festgestellt werden konnte. Nachfolgende drei Richtungen sind möglich:

- Rechte LED leuchtet: Person befindet sich seitlich rechts.
- Beide LED leuchten: Person befindet sich vor dem Roboter.
- Linke LED leuchtet: Person befindet sich seitlich links vom Roboter

Eine höhere Priorität hat die Mitteilung einer zu geringen Akkuspannung, was durch schnelles Blinken beider LEDs signalisiert wird. Beschreibung der LED-Signaldarstellung im Firmware-Update-Modus finden Sie im Abschnitt „Firmware Update durchführen“.

Firmware-Update-Anschluss
Auf der vorderen linken Schulter von Rob-230 befindet sich ein Firmware-Update-Anschluss (5 polige DIN-Buchse) . Diese wird nur verwendet wenn Sie ein Update der Firmware mit dem beiliegenden 5-poligen „Firmware-Update-Aktivierungsstecker“ vornehmen möchten.

Erstmalige Inbetriebnahme

1. Bitte entfernen Sie als erstes vorsichtig sämtliche Verpackungsmaterialien am Roboter, inklusive aller evt. angebrachten Klebebänder.

2. Um ein versehentliches Einschalten des Robotersystems während des Transports zu vermeiden, wurden die elektrischen Verbindungen beider Akkupacks, welche sich auf der Unterseite der beiden Füße befinden, getrennt. Somit öffnen Sie zunächst beide Abdeckungen mit einem Kreuzschraubenzieher und verbinden die beiden Stromstecker. Vergewissern Sie sich bitte zuvor, dass der Betriebsstromschalter (Kippschalter auf der Rückseite) nach unten (= AUS-Stellung) zeigt.

3. Es befindet sich zur Demonstration bereits ein kurzes Steuerprogramm im Flash-Speicher, so dass dieses, nach dem Einschaltvorgang (Kippschalter zeigt nach oben) und einer nachfolgenden kurzen Initialisierungssequenz, sofort ausgeführt wird. Sie können den Roboter zu jeder Zeit, auch im laufenden Betrieb, über diesen Betriebsstromschalter ausschalten.

4. Die Akkupacks wurden bereits ab Werk vorgeladen und müssen, sobald die beiden LEDs auf der Vorderseite im schnellen Rhythmus blinken, für maximal **15 Stunden** erneut aufgeladen werden. Bitte beachten Sie, dass ein Überladen die Lebensdauer der Akkuzellen deutlich herabsetzen kann!

Vorbereitung für die Erstellung eines Steuerprogramms

Um ein Steuerprogramm für Rob-230 mit einem PC zu erstellen, muss zunächst ein Terminalprogramm darauf installiert sein. Auf der beiliegenden CD unter dem Verzeichnis „Terminalprogramm“ finden Sie das Programm „ShamCom“, welches Sie bitte auf Ihrem PC installieren.

Die Kommunikation mit dem Rob-230 System und dem Terminalprogramm erfolgt ausschließlich über eine RS232 Schnittstelle, die allerdings bei modernen Notebooks oft nicht mehr vorhanden ist. Aus diesem Grund gehört zum Lieferumfang auch ein „USB to SERIAL Converter“, den Sie dann zusätzlich noch installieren müssen. Die genaue Installationsanweisung entnehmen Sie bitte der beiliegenden Beschreibung.
.

Konfiguration des Terminalprogramms „ShamCom“

1. Programm „ShamCom“ starten.

2. Name (z.B. „Rob-230“) der neuen zu konfigurierenden seriellen Schnittstelle eingeben.
   Aus der angezeigten Liste der zur Verfügung stehenden korrekte serielle Schnittstelle auswählen.

3. Unter dem Menüpunkt „Optionen“ setzen Sie bitte folgende Einstellungen:

Baudrate:	57600
Stopbits:	1
Datenbits:	8 Bit
Parität:	Keine

4. Drücken Sie auf „OK“ und es öffnet sich die Bedienoberfläche von ShamCom, wo Sie unter dem Menüpunkt „Konfiguration“ und „Terminaleinstellungen..“ folgende Einstellungen vornehmen:

Terminal-Emulation:	VT100/Ansi
Echomodus:	Kein Echo
Zeichencode:	IBM (MSDOS)
Terminalbreite:	80 Zeichen
Empfang von CR:	CR -> CR
<Return>:	Sendet CR

5. Übernahme und Speicherung der neuen Einstellungen mit „OK“

Ausführ-, Programmier- und Firmware-Update-Modus

Es existieren insgesamt drei unterschiedliche Arbeitsmodes für das Rob-230 System, welche nachfolgende Funktionen hat:

Ausführmodus
Wie bereits kurz in dem Abschnitt „Erstmalige Inbetriebnahme“ beschrieben, wird ein im Flash-Speicher abgelegtes Steuerprogramm sofort ausgeführt, sobald Sie den Roboter über den Betriebsstromschalter einschalten.

Programmiermodus
Um in den Programmiermodus für die Neueingabe bzw. Nachbearbeitung eines Steuerprogrammes zu gelangen, muss der Betriebsstromschalter ausgeschaltet, und der Stromstecker des Ladenetzteiles auf der Vorderseite eingesteckt sein. Bei einer zuvor eingerichteten Terminalverbindung wird nach ca. 4 sec der Programmzeileneditor für Ihre Steuerprogrammeingabe dargestellt.

Firmware-Update-Modus
Im Laufe der Zeit wird unsere Firmware stetig erweitert. Um Ihr Robotersystem softwaretechnisch immer auf den neuesten Stand zu halten, können Sie über den „Firmware-Update-Modus“ das System auf einfachste Art und Weise aktualisieren. In diesen Modus gelangen Sie, wenn der Betriebsstromschalter ausgeschaltet ist und der Firmware-Update-Aktivierungsstecker in die auf der Frontseite befindliche 5-polige DIN-Buchse eingesteckt ist. Sobald Sie dann den Stromstecker des Ladenetzteiles in die Strombuchse auf der Vorderseite einstecken, ist das System für ein Firmware-Update über die serielle Schnittstelle aktiviert. Während das System auf einen gültigen Datenstrom wartet, wird dies durch die rechte aktivierte rote LED signalisiert. Weitere Einzelheiten erfahren Sie im Abschnitt „Firmware-Update durchführen“.

Programmzeileneditor

Der im Rob-230 System integrierte Editor ist ein sogenannter Zeileneditor, der die Aufnahme von bis zu 450 Programmzeilen erlaubt. Der Editor lässt sich in folgende drei Bereiche aufteilen, die durch Trennungslinien optisch abgegrenzt sind:

1. Steuerprogramm-Darstellungsfenster (21 Zeilen)
2. Statuszeile (1 Zeile)
3. Eingabezeile (1 Zeile)

Steuerprogramm-Darstellungsfenster
In diesem Bereich werden maximal 21 Zeilen von Ihrem eingegebenen Steuerprogramm dargestellt. Für das zeilenweise Vor- bzw. Zurückscrollen, können Sie die Pfeiltasten <oben> bzw. <unten> oder die Funktionstasten <F1> bzw. <F2> verwenden. Für größere Scroll-Bewegungen (20 Zeilen) verwenden Sie bitte die Funktionstasten <F3> bzw. <F4>.

Statuszeile
Die Statuszeile teilt Ihnen folgende Informationen mit:

- Firmware-Version nach dem Aufruf des Programmiermodus bzw. Editors.
- Fehlerbeschreibung nach der Eingabe einer fehlerhaften Programmzeile.
- Bearbeitungsvorgänge, wie z.B. das Speichern und Laden eines Steuerprogramms.

Eingabezeile
Hier erstellen Sie eine einzelne Steuerprogrammzeile, die Sie auch leicht editieren können. Zum Editieren stehen Ihnen die „Backspace“-Taste (löscht das links vom Cursor stehende Zeichen) und die Pfeiltasten <rechts> bzw. <links> zur Verfügung, womit Sie den Cursor an eine beliebige Position innerhalb der Zeile bewegen können. Die Zeicheneingabe erfolgt immer im Einfügemodus, weshalb evt. rechtsstehende Zeichen nach rechts bezüglich der Cursorposition verschoben werden.

Funktionsweise des Programmzeileneditors

Da der gesamte Editor in der Firmware integriert wurde, welche sich im Mikrocontroller des Rob-230 Systems befindet, musste dieser recht kompakt und somit einfach in seiner Funktionsweise ausgeführt werden. Es stehen für seine Bedienung zwar nur wenige Funktionen zur Verfügung, doch sind diese für die Programmerstellung völlig ausreichend.

- In dieser ersten Firmware-Version kann nur ein Anwenderprogramm bearbeitet und im internen Flash-Speicher des Rob-230 Systems gespeichert werden. Die Möglichkeit ein solches Steuerprogramm auch extern auf einem PC auszulagern wird in einer nachfolgenden Firmware-Version verwirklicht.

- Die maximale Anzahl von Steuerprogrammzeilen beträgt 450 Zeilen.

- Jede Programmzeile die Sie in der Eingabezeile erstellen, muss mit einer **Zeilennummer im Bereich zwischen 1 und 450** beginnen. Nach der Programzeileneingabe wird diese entsprechend der Zeilennummer automatisch im bisher erstellten Steuerprogramm eingefügt.
  Tipps:
  Sie sollten an bestimmten Stellen im Programm bewusst einige Zeilennummern auslassen, um auch nachträglich noch Programmzeilen einfügen zu können.
  Sie können die Übersichtlichkeit von zusammenhängenden Programmabschnitten deutlich erhöhen, wenn Sie zwischen den einzelnen Programmabschnitten „Leerzeilen“ einfügen.

- Das Löschen einer bereits eingegebenen Programmzeile, erfolgt durch Eingabe der entsprechenden Zeilennummer, ohne jede nachfolgende Zeicheneingabe. Nach Betätigung der „Enter“-Taste wird diese sofort entfernt.

- Ein Überschreiben einer bereits vorhandenen Steuerprogrammzeile wird durch Neueingabe mit identischer Zeilennummer erreicht.

Editor- und Ausführbereich

Das Steuerprogramm, welches Sie mit dem Editor erstellen, befindet sich zunächst in einem flüchtigen Speicher, welches sofort verloren geht, wenn die Stromversorgung unterbrochen wird. Somit muss spätestens, wenn Sie die Programmierung beenden möchten, das zuvor im Editor erstellte Programm in den nicht flüchtigen Flash-Speicher für den Ausführbereich kopiert bzw. gesichert werden (Ctrl S). Nur ein Steuerprogramm, welches sich im Ausführbereich befindet wird auch, nachdem der Betriebsstromschalter eingeschaltet wurde, ausgeführt.
Es wird empfohlen auch „zwischenzeitlich" während der Programmerstellung stets eine Sicherung vorzunehmen.
Umgekehrt, wenn Sie ein im Flash-Speicher bereits abgelegtes Steuerprogramm weiter bearbeiten möchten, muss dieses zunächst in den Editor mit „Ctrl L" geladen werden
.
Immer zu Beginn, wenn Sie den Programmiermodus starten (Stromstecker des Ladeadapters in die auf der Vorderseite angebrachte Strombuchse stecken), ist der Speicher des Editors komplett gelöscht.

Zusammenfassung der Editorbefehle

Pfeiltaste <oben> bzw. Funktionstaste <F1>
Scrollt das Anwenderprogramm im Darstellungsfenster um eine Programmzeile nach oben.

Pfeiltaste <unten> bzw. Funktionstaste <F2>
Scrollt das Anwenderprogramm im Darstellungsfenster um eine Programmzeile nach unten.

Funktionstaste <F3>
Scrollt das Anwenderprogramm im Darstellungsfenster 20 Programmzeilen nach oben.

Funktionstaste <F4>
Scrollt das Anwenderprogramm im Darstellungsfenster 20 Programmzeilen nach unten.

Taste <Backspace>
Löscht das links vom Cursor stehende Zeichen

Pfeiltaste <rechts>
Bewegt den Cursor um eine Stelle nach rechts auf der Eingabezeile.

Pfeiltaste <links>
Bewegt den Cursor um eine Stelle nach links auf der Eingabezeile.

Tastenkombination: Ctrl S
Speichert bzw. kopiert das im Editor befindliche Steuerprogramm in den Flash-Speicher des Ausführungsbereiches, welches erst dann vom Robotersystem ausgeführt werden kann. Der Speichervorgang kann einige Sekunden dauern, wobei Sie entsprechend in der Statuszeile informiert werden. Bitte unterbrechen Sie während dieses Speichervorganges NICHT die Stromversorgung!

Tastenkombination: Ctrl L
Lädt ein im Flash-Speicher zuvor abgelegtes Steuerprogramm in den Editor, welches dann weiter bearbeitet werden kann.

Tastenkombination: Ctrl B
Verzweigt in ein Utility-Menü das die Hauptaufgabe hat, die von den Sensoren gemessenen Werte und den empfangenen Code (Remote-Code) der IR-Fernbedienung darzustellen. Neben einer allgemeinen Funktionsprüfung können Sie diese Werte und Wertebereiche in Ihrem Steuerprogramm unter Anderem dazu. verwenden einen Schwellwert zu berechnen um bei Überschreitung eine bestimmte Roboteraktion auszulösen.

System Utility Menü

Über dieses Menü können Sie Werte der Sensoren und Remote-Codes der IR-Fernbedienung in Echtzeit darstellen, Sprachbefehle für die Spracherkennung trainieren, sowie englische Sätze für die synthetische Sprachausgabe eingeben.
Das Menü wird aufgerufen, wenn Sie sich im Programmiermodus befinden und die Tastenkombination Ctrl B drücken. Über die Taste <ESC> verlassen Sie das Utility Menü und gelangen zurück in den Editorbereich.

Die Werte folgender Sensoren bzw. Datenerfassungssysteme werden nach Betätigung der entsprechenden Auswahltaste angezeigt:

<1> Ausgabe der Versorgungsspannung des eingesteckten Ladenetzteils

<2> Geräuschmessung der Umgebungslautstärke. Es werden im zeitlichen Abstand von 0,3 sec fortlaufend 20 Messungen durchgeführt und zeilenweise dargestellt. Abschließend wird nach 3 sec automatisch wieder in das Utility Menü zurückgekehrt.

<3> Entfernungsmessung des Ultraschallsensors. Es werden im zeitlichen Abstand von 0,3 sec fortlaufend 20 Messungen durchgeführt und zeilenweise dargestellt. Abschließend wird nach 3 sec automatisch wieder in das Utility Menü zurückgekehrt.

<4> Helligkeitsmessung der beiden Lichtsensoren. Es werden im zeitlichen Abstand von 0,3 sec fortlaufend 20 Messungen vom rechten und linken Lichtsensor durchgeführt und zeilenweise dargestellt. Hierbei entsprechen die Werte der rechten Spalte dem rechten Lichtsensor. Abschließend wird nach 3 sec automatisch wieder in das Utility Menü zurückgekehrt.

<5> Personenortung über die beiden IR-Passiv-Sensoren. Es werden im zeitlichen Abstand von 0,3 sec fortlaufend 20 Messungen vom rechten und linken Sensor durchgeführt und zeilenweise dargestellt. Hierbei entsprechen die Werte der rechten Spalte dem rechten IR-Passiv-Sensor. Abschließend wird nach 3 sec automatisch wieder in das Utility Menü zurückgekehrt.

<6> Darstellung der empfangenen Remote-Codes der IR-Fernbedienung. Jeder Tastendruck auf der Fernbedienung zeigt dessen zugehörigen Remote-Code zeilenweise an. Nach dem Empfang des 20. Remote-Codes wird automatisch wieder in das Utility Menü zurückgekehrt.

WICHTIGER HINWEIS:
Um den Empfang von Remote-Codes zu ermöglichen, muss nachdem Sie das Utility Menü aufgerufen haben, der Betriebsstromschalter eingeschaltet werden, da die Stromversorgung des IR-Empfängers nur über den Akku versorgt wird.

<A> Verzweigt in das Untermenü „Utility-Menü: Sprachausgabemodul“ für die Eingabe englischer Sätze, die Sie in Ihrem Steuerprogramm über das integrierte Sprachausgabemodul (Text to Speech) ausgeben können.

<B> Verzweigt in das Untermenü „Utility Menü: Spracherkennungsmodul“ für die Erstellung (inkl. Training) von Sprachbefehlen, die Sie in Ihrem Steuerprogramm abfragen können.

<ESC> Verlassen des System Utility Menü und Rückkehr in den Editorbereich.

Utility Menü: Sprachausgabemodul

Es stehen folgende Menüpunkte zur Auswahl:

<1> Eingabe einer englischen Textzeile die eine Länge von bis zu 40 Zeichen betragen darf, inkl. einer vom Anwender festgelegten Index-Nr. Unter dieser Index-Nr wird im Steuerprogramm der Text ausgewählt und über den eingebauten Lautsprecher ausgegeben.

<2> Auflistung der bisher gespeicherten Texte mit der zugehörigen Index-Nr.

<3> Akustische Ausgabe eines an dieser Stelle eingegebenen Textes, der jedoch nicht gespeichert wird. Diese Funktion dient lediglich dazu einen Satz akustisch zu überprüfen (Testausgabe).

<Z> Löschen des gesamten Textspeichers.

<ESC> Rückkehr in das „System Utility Menü“.

Utility Menü: Spracherkennungsmodul

Es stehen folgende Menüpunkte zur Auswahl:

<1> Einen Zuruf trainieren und speichern. Der Zuruf muss zweimal hintereinander mit gleicher Klangfarbe und Lautstärke gesprochen werden. Beachten Sie bitte hierzu die in deutsch gesprochenen Anweisungen. Nach erfolgreichem Training eines Zurufs wird vom System eine Index-Nr beginnend mit 1 erzeugt, die sich der Anwender unbedingt merken sollte. Diese Index-Nr kann der Anwender später in seinem Steuerprogramm nach einem erkannten Zuruf abfragen und somit zuordnen.

<2> Mit dieser Funktion kann ein Zuruf auf Erkennung getestet werden.

<Z> Löscht alle bisher gespeicherten Zurufe mit den zugehörigen Index-Nr. Wird ein neuer Zuruf erstellt, so beginnt dieser wieder mit der index-Nr 1.

<ESC Rückkehr in das „System Utility Menü“.

Programmiersprache für die Steuerprogrammerstellung

Für das Rob-230 System wurde eine ganz spezielle Programmiersprache entwickelt, die eine schnelle Einarbeitung ermöglicht und trotzdem alle Funktionen des Roboters hervorragend unterstützt. Es ist ein Interpreter, der ein wenig einem Basic-Dialekt ähnelt. Es wurden allerdings nur solche Funktionen implementiert, die für eine effektive und sinnvolle Programmerstellung ausreichend sind. Durch die Verwendung eines Interpreters können Sie sofort nach der Steuerprogrammeingabe die Aktionen des Roboters im Ausführungsmodus beobachten.

Grundlegender Aufbau einer Programmzeile
Jede Programmzeile kann bis zu 4 Einträge aufweisen, die spaltenweise mit mindestens einem Leerzeichen voneinander getrennt sein müssen.

1. Der erste Eintrag enthält immer die Zeilennummer, die zwischen 1 und 450 liegen muss.

2. Der zweite Eintrag spezifiziert den Befehl, auf dessen Groß- und Kleinschreibung nicht geachtet werden muss. Der Editor konvertiert ggf. jede eingegebene Programmzeile automatisch auf Kleinschreibung um. Für ein besseres Verständnis sind fast alle Befehle aus mehreren Wörtern zusammengesetzt, die mit einem Unterstrich, wie z.B. beim Befehl „left_arm_up“ , miteinander verbunden sind.

3. Abhängig vom Befehl sind entweder keine weiteren Einträge mehr notwendig, oder ein weiterer dritter Parametereintrag, der dann als Integer-Variable, Konstante oder Zeilennummer definiert ist.

4. Maximal ist auch noch ein vierter Parametereintrag möglich, der eine Integer-Variable oder eine Zeilennummer spezifiziert.

Beispiele für mögliche Steuerprogrammzeilen

10 left_arm_up
Um nur den linken Arm anzuheben sind lediglich 2 Einträge für Zeilennummer und Befehl notwendig.

30 walk_forward_time 40
Drei Einträge sind notwendig, damit sich der Roboter in Vorwärtsrichtung für 4 Sekunden bewegt. Die Parameterangabe im dritten Eintrag entspricht der Zeit in zehntel Sekunden, und entspricht in diesem Beispiel der Konstante „40“.

122 copy 3491 int23
Um einen Wert von z.B. die Integer-Konstante 3491 in eine Integer-Variable int23 zu kopieren sind 4 Einträge notwendig.

95 if_int31_not_equal_goto int3 200
Bei einem Vergleich wird der Inhalt der Integer-Variable int3 mit dem Inhalt der Variablen int31 verglichen. Ein Sprung zur Programmzeile „200“ wird nur dann ausgeführt, wenn die Werte in den beiden Variablen ungleich sind.

Anwender- und Systemvariable

Für den Anwender stehen 30 Variable (int1 bis int30) für sein erstelltes Steuerprogramm zur Verfügung. Diese sind alle vom Typ integer und decken somit den Wertbereich von -32768 bis +32767 ab. Die restlichen Variablen (int31 bis int54), sind ebenfalls vom Typ integer und sind die sogenannten Systemvariablen, die speziell für das System reserviert sind, das darin z.B. seine Sensorwerte speichert, die der Anwender dann in seinem Steuerprogramm auslesen und weiterverarbeiten kann. In anderen Systemvariablen können wiederum bestimmte Einstellungen vom Anwender vorgenommen werden, die das System dann während seiner Programmausführung berücksichtigt. Eine ausführliche Beschreibung über die unterschiedlichen Funktionen der zur Verfügung stehenden Systemvariablen ist nachfolgend aufgeführt.

Anwendervariable

int1, int2, ... , int30
Diese insgesamt 30 Anwendervariablen stehen für den Anwender in seinem Steuerprogramm zur freien Verwendung.

Systemvariable

int31
Möchten Sie zwei Werte miteinander vergleichen (Compare-Befehl), so muß zunächst diese Systemvariable mit einem der beiden Werte geladen werden. Der andere Wert wird in der Compare-Befehlszeile spezifiziert. Siehe auch hierzu die Beschreibung der Compare-Befehle.

int32
Die Systemvariable int32 enthält den Wert der Akkuspannung. Dabei entspricht der Wert 65 etwa 1V.

int33
Die Systemvariable int33 können Sie mit einem bestimmten Spannungspegelwert laden. Wird dieser Pegelwert von dem Wert der aktuellen Akkuspannung (int32) unterschritten, so wird eine Flag-Variable (int34) auf 1 gesetzt und ein dauerhaftes Warnsignal (schnelles Blinken beider LEDs) aktiviert.

int34
Dies ist eine sogenannte Flag-Variable und wird nach dem Start von Rob-230 mit 0 initialisiert. Unterschreitet die Akkuspannung den in der Systemvariable int33 gesetzten Wert, so wird diese Flag-Variable auf 1 gesetzt.

int35
Enthält die Beleuchtungsstärke des rechten Lichtsensor (Fotowiderstand). Dieser Wert ist weder genormt noch linear bezüglich der Beleuchtungsstärke. Der theoretische Wertbereich kann zwischen 0 und 1023 liegen.

int36
Enthält die Beleuchtungsstärke des linken Lichtsensor (Fotowiderstand). Dieser Wert ist weder genormt noch linear bezüglich der Beleuchtungsstärke. Der theoretische Wertbereich kann zwischen 0 und 1023 liegen.

int37
Enthält den Wert der momentanen Lautstärke. Dieser Wert ist weder genormt noch linear und kann theoretisch zwischen 0 und 1023 liegen.

int38
Die Systemvariable int38 können Sie mit einem bestimmten Lautstärkepegelwert laden. Wird dieser Pegelwert von der aktuellen Lautstärke (int37) überschritten, so wird eine Flag-Variable (int39) auf 1 gesetzt.

int39
Diese Flagvariable wird nach dem Start von Rob-230 mit 0 initialisiert. Überschreitet die aktuelle Lautstärke den in der Systemvariable int38 gesetzten Lautstärkepegel, so wird die Flag-Variable auf 1 gesetzt.

int40
Enthält den aktuellen Pegelwert des rechten Passiv-Infrarot-Sensor. Der theoretische Wertebereich kann zwischen 0 und 1023 liegen. Im Ruhezustand (Sensor registriert keine Bewegung) wird ein Zahlenwert von ungefähr 512 ausgegeben. Durch Anregung des Sensors (eine Person bewegt sich) wird ausgehend vom Ruhezustand sowohl ein positiver (int40 > 512), wie auch ein negativer Pegelwert (int40 < 512) generiert. Der Wert welcher vom Sensor generiert wird, pendelt dynamisch um den Wert des Ruhezustandes.

int41
Enthält den aktuellen Pegelwert des linken Passiv-Infrarot-Sensor. Der theoretische Wertebereich kann zwischen 0 und 1023 liegen. Im Ruhezustand (Sensor registriert keine Bewegung) wird ein Zahlenwert von ungefähr 512 ausgegeben. Durch Anregung des Sensors (eine Person bewegt sich) wird ausgehend vom Ruhezustand sowohl ein positiver (Int40 > 512), wie auch ein negativer (int40 < 512) Pegelwert generiert. Der Wert welcher vom Sensor generiert wird, pendelt dynamisch um den Wert des Ruhezustandes.

int42
Mit dieser Systemvariable bestimmen Sie einen Pegelwert, der bei positiver Überschreitung (ausgehend vom Ruhezustand des Passiv-IR-Sensors) durch ein Passiv-Infrarot-Signal, als „Bewegung registriert“ gezählt und vom System ausgewertet wird. Sie können also die Empfindlichkeit für das Erkennen von bewegenden Personen und dessen Richtung somit erhöhen bzw. verkleinern. Der theoretische Wertebereich kann somit zwischen 513 und 1023 liegen. Bitte beachten Sie, dass eine Veränderung dieses Wertes nur dann sinnvoll ist, wenn Sie gleichzeitig auch den Wert in Systemvariable int43 im gleichen Maße (nur in entgegengesetzter negativer Pegelrichtung) verändern.

int43
Mit dieser Systemvariable bestimmen Sie einen Pegelwert, der bei negativer Überschreitung (ausgehend vom Ruhezustand des Passiv-IR-Sensors) durch ein Passiv-Infrarot-Signal, als „Bewegung registriert“ gezählt und vom System ausgewertet wird. Sie können also die Empfindlichkeit für das Erkennen von bewegenden Personen und dessen Richtung somit erhöhen bzw. verkleinern. Der theoretische Wertebereich kann somit zwischen 511 und 0 liegen. Bitte beachten Sie, dass eine Veränderung dieses Wertes nur dann sinnvoll ist, wenn Sie gleichzeitig auch den Wert in Systemvariable int42 im gleichen Maße (nur in entgegengesetzter positiver Pegelrichtung) verändern.

int44
Für eine sichere Richtungsangabe einer bewegenden Person, wird vom System eine gewisse Richtungs-Registrierungszeit (Voreingestellt sind ca. 2 sec) veranschlagt. In dieser Zeit müssen beide Passiv-IR-Sensoren ein konstantes Signal (Person bewegt sich bzw. Person bewegt sich nicht) liefern. Wird von einem Passiv-IR-Sensor eine bewegende Person registriert, so wird dieses Ergebnis eine Zeitlang aufrechterhalten, auch wenn danach kurzzeitig keine Bewegung mehr erkannt wurde. Diese maximale Zeitspanne wird in der Systemvariable int44 hinterlegt. Im Allgemeinen ist keine Veränderung des voreingestellten Wertes notwendig. Doch können Sie ruhig mit unterschiedlichen Werten experimentieren, um somit gewisse Veränderungen bei der Richtungsermittlung durch Rob-230 zu beobachten.

int45
Für eine sichere Richtungsangabe einer bewegenden Person, wird vom System eine gewisse Richtungs-Registrierungszeit veranschlagt. In dieser Zeit müssen beide Passiv-IR-Sensoren
ein konstantes Signal (Person bewegt sich bzw. Person bewegt sich nicht) liefern. Die Systemvariable int45 enthält diese Richtungs-Registrierungszeit, welche auf ca. 2 sec voreingestellt ist. Je länger diese Zeit gewählt wird, desto sicherer kann eine bestimmte Richtungsaussage vom System gemacht werden. Gleichzeitig jedoch nimmt die Trägheit für eine Richtungsaussage zu. D.h. wenn sich die Richtung einer Person zu schnell ändert, wird das System auch keine Richtungsaussage mehr machen können (int46 = 0).

int46
Diese Systemvariable enthält die von Rob-230 registrierte Richtung einer Person.
Folgende Werte sind möglich:

0 = Es wurde keine eindeutige Richtung einer Person festgestellt
1 = Person befindet sich seitlich rechts von Rob-230
2 = Person befindet sich direkt vor Rob-230
3 = Person befindet sich seitlich links von Rob-230

int47
Bewegungen von Personen können nur registriert werden, wenn sich das System selbst in einem Ruhezustand befindet. Auch nachdem ein Bewegungsvorgang beendet wurde muss eine bestimmte Zeit gewartet werden, bis die Passiv-IR-Sensoren (Bewegungsmelder) für Messungen aktiviert werden können. Diese sogenannte Totzeit nach Beendigung eines Bewegungsvorganges ist in der Systemvariable int47 hinterlegt. Voreingestellt ist eine Totzeit von 2,4 sec, was einem Wert von 60 entspricht. Die zeitliche Auflösung beträgt somit 0,04 sec.

int48
Wird ein Bewegungsvorgang vorzeitig durch Über- bzw. Unterschreitung einer Ultraschallentfernungsmessung gestoppt, so kann diese Bewegung eine gewisse Zeit, die die Systemvariable int48 enthält, fortgesetzt werden. Nach dem Einschaltvorgang wird diese Systemvariable mit 0 initialisiert. Die zeitliche Auflösung beträgt 0,04 sec, womit int48 = 25 einer Zeit von 1 sec entspricht.

Int49
Einstellung der Lautstärke für das synthetische Sprachausgabemodul. Der gültige Wertbereich liegt zwischen 0 und 7, wobei der Wert 0 der maximalen Lautstärke entspricht. Voreingestellt ist der Wert 0.

Int50
Einstellung der Geschwindigkeit der Sprachausgabe. Der gültige Wertbereich liegt zwischen 0 und 7, wobei der Wert 0 der minimalsten Ausgabegeschwindigkeit entspricht. Voreingestellt ist der Wert 4.

Int51
Einstellung der Tonlage für das synthetische Sprachausgabemodul. Voreingestellt ist der Wert 2.

Int52
Nachdem vom Spracherkennungsmodul mit dem Befehl „notice_recognition_time_goto“ ein Zuruf registriert worden ist, steht in dieser Systemvariable bei erkanntem Zuruf die zugehörige Indexnummer 0 – 59. Konnte jedoch der Zuruf nicht erkannt werden, so steht ein entsprechender Fehlercode beginnend ab 128 in dieser Systemvariable.

int53
Reserviert.

int54
Wird während dem Betrieb ein Remote-Code von der Fernbedienung empfangen, so steht in dieser Systemvariable der zuletzt empfangene Code.

Steuerprogrammbefehle

Es lassen sich folgende Typen von Befehlen unterscheiden:

- Befehle für zeitabhängige Geh- und Drehbewegungen.
- Befehle für zeitabhängige Geh- und Drehbewegungen mit vorzeitiger Abbruchmöglichkeit.
- Befehle für die Bewegung der Arme und des Oberkörpers.
- Befehle für die Ausführung einer kompletten Bewegungssequenz mit Soundausgabe.
- Befehle für die Registrierung und Auswertung von Sensorsignalen.
- Befehle für das Setzen von Parametern.
- Allgemeine Befehle der Programmsteuerung.
- Transferbefehle.
- Arithmetische Befehle.
- Befehle um Werte zu vergleichen.
- Befehle für die Ein- und Ausgabe über die serielle RS232 Schnittstelle.
- Befehle für die Ansteuerung des Sprachausgabemoduls.
- Befehl für die Ansteuerung des Spracherkennungsmoduls.
- Sonstige Befehle

Befehle für zeitabhängige Geh- und Drehbewegungen

Befehl	**Beschreibung**
walk_forward_time	Bewegung geradeaus in zehntel Sekunden Parameterübergabe: Konstante/Variable
walk_backward_time	Bewegung rückwärts in zehntel Sekunden Parameterübergabe: Konstante/Variable
turn_right_time	Drehung rechts auf der Stelle in zehntel Sekunden Parameterübergabe: Konstante/Variable
turn_left_time	Drehung links auf der Stelle in zehntel Sekunden Parameterübergabe: Konstante/Variable

Befehle für zeitabhängige Geh- und Drehbewegungen mit vorzeitiger Abbruchmöglichkeit

Nachfolgend sind alle zeitlich einstellbaren Geh- und Drehbewegungen aufgeführt, dessen Bewegungsvorgang vorzeitig abgebrochen werden kann, wenn die Entfernungsmessung mit dem Ultraschallsensor einen zuvor festgelegten Entfernungswert unter- bzw. überschreitet.

Befehl	**Beschreibung**
walk_forward_time_near	Bewegung geradeaus in zehntel Sec mit vorzeitigem Bewegungsstop, wenn die Entfernung zu einem vorausstehenden Objekt zu gering ist. Mögliche Anwendung: Stop vor einem Hindernis. Parameterübergabe: 1.Konstante/Variable 2. Zeilennummer
walk_backward_time_far	Bewegung rückwärts in zehntel Sec mit vorzeitigem Bewegungsstop, wenn ein vorausstehendes Objekt zu weit entfernt ist. Mögliche Anwendung: Abstandhalten vor einem sich nähernden Objekt/Person. Parameterübergabe: 1.Konstante/Variable 2. Zeilennummer
turn_right_time_near	Drehbewegung rechts in zehntel Sec mit vorzeitigem Bewegungsstop, wenn die Entfernung zu einem vorausstehenden Objekt zu gering ist. Mögliche Anwendung: Zuwendung einem Objekt bzw. einer Person. Parameterübergabe: 1.Konstante/Variable 2. Zeilennummer
turn_right_time_far	Drehbewegung rechts in zehntel Sec mit vorzeitigem Bewegungsstop, wenn ein vorausstehendes Objekt zu weit entfernt ist. Mögliche Anwendung: Hindernisfreien Durchgang finden. Parameterübergabe: 1.Konstante/Variable 2. Zeilennummer
turn_left_time_near	Drehbewegung links in zehntel Sec mit vorzeitigem Bewegungsstop, wenn die Entfernung zu einem vorausstehenden Objekt zu gering ist. Mögliche Anwendung: Zuwendung einem Objekt bzw. einer Person. Parameterübergabe: 1.Konstante/Variable 2. Zeilennummer
turn_left_time_far	Drehbewegung links in zehntel Sec mit vorzeitigem Bewegungsstop, wenn ein vorausstehendes Objekt zu weit entfernt ist. Mögliche Anwendung: Hindernisfreien Durchgang finden. Parameterübergabe: 1.Konstante/Variable 2. Zeilennummer

Befehle für die Bewegung der Arme und des Oberkörpers

Befehl	**Beschreibung**
right_arm_up left_arm_up	Bewegt den rechten bzw. linken Oberarm um eine Stufe nach oben. Es gibt insgesamt zwei Stufen für die Oberarmstellung. Parameterübergabe: Keine
right_arm_down left_arm_down	Bewegt den rechten bzw. linken Oberarm um eine Stufe nach unten. Es gibt insgesamt zwei Stufen für die Oberarmstellung. Parameterübergabe: Keine
right_arm_out left_arm_out	Bewegt den rechten bzw. linken Unterarm um eine Stufe nach außen. Es gibt insgesamt zwei Stufen für die Unterarmstellung. Parameterübergabe: Keine
right_arm_in left_arm_in	Bewegt den rechten bzw. linken Unterarm um eine Stufe nach innen. Es gibt insgesamt zwei Stufen für die Unterarmstellung. Parameterübergabe: Keine
tilt_body_right tilt_body_left	Bewegt den Oberkörper kurzzeitig nach rechts bzw. nach links. Parameterübergabe: Keine
arms_out	Bewegt beide Arme nach vorne, und kurzzeitig auch den Oberkörper. Parameterübergabe: Keine
arms_in	Bewegt beide Arme nach hinten, und kurzzeitig auch den Oberkörper. Parameterübergabe: Keine

Befehle für die Ausführung einer kompletten Bewegungssequenz mit Soundausgabe

Es werden unterschiedliche Bewegungssequenzen mit entsprechender Soundunterlegung ausgeführt.
Parameterübergabe: Keine

Befehle	**Beschreibung**
right_hand_thump left_hand_thump	Mit der rechten bzw. linken Hand schlagen.
right_hand_throw left_hand_throw	Werfen eines Gegenstandes.
right_hand_pick_up left_hand_pick_up	Aufnehmen eines Gegenstandes.
right_hand_sweep left_hand_sweep	Mit der rechten bzw. linken Hand winken.
right_hand_strike_1 left_hand_strike_1	Kampfsequenz Nr.1 rechts bzw. links.
right_hand_strike_2 left_hand_strike2	Kampfsequenz Nr .2 rechts bzw. links.
right_hand_strike_3 left_hand_strike_3	Kampfsequenz Nr.3 rechts bzw. links.
arms_right_down arms_left_down	Beide Arme werden kurzzeitig nach rechts bzw. links unten gestreckt.
animation_burp	Animation: Rülpsen
animation_oops	Animation: Pupsen
animation_roar	Animation: Grölen
animation_whistle	Animation: Pfeifen
animation_talk_back	
animation_dance	Aufwändiges Tanzdemo mit Soundausgabe

Befehle für die Registrierung und Auswertung von Sensorsignalen

Befehle	Beschreibung
notice_noise_time_goto	Es wird innerhalb der angegebenen Zeit (zehntel sec) auf eine Geräuschpegelüberschreitung gewartet. Wurde während dieser Zeit der Geräuschpegel überschritten, so wird in die angegebene Programmzeile gesprungen. Der zum Vergleich herangezogene Pegelwert steht in der Systemvariablen int38 welcher nach Programmstart mit dem Wert 300 initialisiert wird. Parameterübergabe: 1.Konstante/Variable 2.Zeilennumer
notice_human_time_goto	Es wird innerhalb der angegebenen Zeit (zehntel sec) gewartet die Richtung einer Person festzustellen. Konnte innerhalb dieser Zeit die Richtung eindeutig festgestellt werden, so wird in die angegebene Programmzeile gesprungen. Hinweis: Die Person muss sich zumindest etwas bewegen, damit die IR-Passiv-Sensoren dies auch registrieren können. Der entsprechende Richtungswert befindet sich danach in der Systemvariable Int46 und kann vom Anwenderprogramm ausgewertet werden. Es sind folgende Richtungswerte (in Blickrichtung des Roboters) möglich: 0 = Keine eindeutige Richtung (Es erfolgt keine Programmverzweigung) 1 = Person befindet sich seitlich vorne rechts 2 = Person befindet sich direkt voraus 3 = Person befindet sich seitlich vorne links Parameterübergabe: 1.Konstante/Variable 2.Zeilennummer
notice_remote_time_goto	Es wird innerhalb der angegebenen Zeit (zehntel sec) auf einen gesendeten Remote-Code von der Fernbedienung gewartet. Wurde zu dieser Zeit ein Code empfangen, wird er in der Systemvariable int54 gespeichert und in die angegebene Programmzeile gesprungen. Bei Aufruf dieses Befehls wird int54 zunächst mit 0 initialisiert. Parameterübergabe: 1.Konstante/Variable 2.Zeilennummer
get_sonar_distance	Es wird eine Entfernungsmessung mit dem Ultraschallsensor durchgeführt und der gemessene Wert (cm) in der angegebenen Variable gespeichert. Parameterübergabe: Variable

Befehle für das Setzen von Parametern

Befehle	Beschreibung
set_sonar_distance_near	Setzt einen minimalen Entfernungswert in cm, ab dem ein Bewegungsvorgang bei Unterschreitung bereits vor Ablauf der Bewegungszeit beendet werden kann. Voreingestellt ist eine Entfernung von 40 cm. Folgende Befehle werten diesen Parameter aus: • walk_forward_time_near_goto • turn_right_time_near_goto • turn_left_time_near_goto Parameterübergabe: Konstante/Variable
set_sonar_distance_far	Setzt einen maximalen Sonar-Entfernungswert in cm, ab dem ein Bewegungsvorgang bei Überschreitung bereits vor Ablauf der Bewegungszeit beendet werden kann. Voreingestellt ist eine Entfernung von 80 cm. Folgende Befehle werten diesen Parameter aus: • walk_backward_time_far_goto • turn_right_time_far_goto • turn_left_time_far_goto Parameterübergabe: Konstante/Variable

Allgemeine Befehle der Programmsteuerung

Befehle	Beschreibung
goto	Führt einen Sprung zu der angegebene Programmzeile durch. Parameterübergabe: Zeilennummer
gosub	Vom Hauptprogramm aus wird ein Unterprogramm, welches sich an der angegebenen Programmzeile befindet, aufgerufen. Das Unterprogramm muss am Ende mit dem Befehl Return abgeschlossen werden. Danach wird die nächste Programmzeile, nach dem „gosub“-Befehl, fortgesetzt. Aus einem Unterprogramm heraus darf kein weiterer Unterprogrammaufruf mehr erfolgen. Parameterübergabe: Zeilennummer
return	Mit diesem Befehl wird ein zuvor vom Hauptprogramm (durch den Befehl „gosub“) aufgerufenes Unterprogramm beendet. Danach wird die Programmausführung, nach dem „gosub“-Befehl, fortgesetzt. Parameterübergabe: Keine
wait	Die Programmausführung wartet um die entsprechend angegebene Zeit in zehntel sec. Parameterübergabe: Konstante/Variable
wait_sec	Die Programmausführung wartet um die entsprechend angegebene Zeit in sec. Parameterübergabe: Konstante/Variable

Transferbefehle

Befehl	Beschreibung
copy	Dieser Befehl kopiert den Wert einer Konstanten oder den Inhalt einer Variable in eine weitere Variable. Es können sowohl Anwender- wie auch Systemvariable damit bearbeitet werden. Parameterübergabe: 1.Konstante/Variable 2.Variable

Arithmetische Befehle

Befehl	Beschreibung
add	Addiert den Wert einer Konstanten oder den Inhalt einer Variablen zu dem Inhalt einer weiteren Variable. Parameterübergabe: 1.Konstante/Variable 2.Variable
sub	Subtrahiert den Wert einer Konstanten oder den Inhalt einer Variablen von dem Inhalt einer weiteren Variable. Parameterübergabe: 1.Konstante/Variable 2.Variable
mul	Multipliziert den Wert einer Konstanten oder den Inhalt einer Variablen mit dem Inhalt einer weiteren Variablen. Parameterübergabe: 1.Konstante/Variable 2.Variable
div	Dividiert den Wert einer Konstanten oder den Inhalt einer Variablen durch den Inhalt einer weiteren Variable. Das Ergebnis wird in der 2. Variablen gespeichert. Parameterübergabe: 1.Konstante/Variable 2.Variable

Befehle um Werte zu vergleichen

Für den Vergleich von 2 Werten, muss zunächst die Systemvariable Int31 mit dem 1. Vergleichswert geladen werden. Der 2. Wert kann eine Konstante oder Variable sein, welche als 1. Parameter in der Programmzeile angegeben wird. Der 2. Parameter enthält die Zeilennummer, zu der gesprungen wird, wenn der entsprechende logische Vergleich zwischen dem 1. Parameter mit dem Wert in der Systemvariable int31 wahr ist.

Befehl	Beschreibung
if_int31_less_equal_goto	Sprung wird ausgeführt wenn der Wert in der Systemvariable int31 kleiner oder gleich ist, wie der in der Steuerprogrammzeile angegebene Parameterwert. Parameterübergabe: 1.Konstante/Variable 2.Zeilennummer
if_int31_less_goto	Sprung wird ausgeführt wenn der Wert in der Systemvariable int31 kleiner ist, als der in der Steuerprogrammzeile angegebene Parameterwert. Parameterübergabe: 1.Konstante/Variable 2. Zeilennummer
if_int31_greater_equal_goto	Sprung wird ausgeführt wenn der Wert in der Systemvariable int31 größer oder gleich ist, wie der in der Steuerprogrammzeile angegebene Parameterwert. Parameterübergabe: 1.Konstante/Variable 2. Zeilennummer
if_int31_greater_goto	Sprung wird ausgeführt wenn der Wert in der Systemvariable int31 größer ist, als der in der Steuerprogrammzeile angegebene Parameterwert. Parameterübergabe: 1.Konstante/Variable 2. Zeilennummer
if_int31_equal_goto	Sprung wird ausgeführt wenn der Wert in der Systemvariable int31 gleich ist, wie der in der Steuerprogrammzeile angegebene Parameterwert. Parameterübergabe: 1.Konstante/Variable 2. Zeilennummer
if_int31_not_equal_goto	Sprung wird ausgeführt wenn der Wert in der Systemvariable int31 ungleich ist, wie der in der Steuerprogrammzeile angegebene Parameterwert. Parameterübergabe: 1.Konstante/Variable 2. Zeilennummer

Befehle für die Ein- und Ausgabe über die serielle Schnittstelle

Befehl	Beschreibung
print	Der Wert einer Variablen oder Konstante wird als Zeichenkette an die serielle Schnittstelle ausgegeben. Bitte beachten Sie, wenn Sie ein Terminal angeschlossen haben, daß nach Ausgabe dieses Befehls keine neue Zeile begonnen wird, wie dies bei den meisten Programmiersprachen üblich ist. Um eine neue Zeile zu beginnen senden Sie im Anschluss nacheinander die Befehle „print_bin 13“ und „print_bin 10“ Parameterübergabe: Konstante/Variable
print_bin	Der Wert einer Variablen oder Konstante wird direkt als Byte an die serielle Schnittstelle gesendet. Somit bestünde die Möglichkeit auch extern angeschlossene Hardware, die über eine RS232-Schnittstelle verfügen, anzusteuern. Parameterübergabe: Konstante/Variable
input_bin	Dieser Befehl wartet auf ein Byte, welches über die serielle Schnittstelle empfangen wird. Ist ein Terminal angeschlossen, so wartet die Programmausführung, bis eine Taste gedrückt wurde. Der empfangene Byte-Wert wird in der mit angegebenen Variable gespeichert. Parameterübergabe: Variable

Befehle für die Ansteuerung des Sprachausgabemoduls

Befehle	Beschreibung
talk_text_number	Der unter der Index-Nr im Utility Menü gespeicherte Text wird über das synthetische Sprachausgabemodul ausgegeben. Parameterübergabe: Konstante/Variable
wait_talk_end	Die weitere Programmausführung wartet, bis die Sprachausgabe beendet wurde. Parameterübergabe: Keine

Befehl für die Ansteuerung des Spracherkennungsmoduls

Befehle	Beschreibung
notice_recognition_time_goto	Der erste Parameter enthält die Registrierungszeit in zehntel Sekunden, indem die weitere Programmausführung auf einen Zuruf vom Anwender wartet. Dieser Zuruf muss zuvor im Utility Menü trainiert und gespeichert worden sein, wobei das System gleichzeitig eine zugehörige Index-Nr vergibt. Wurde innerhalb der Registrierungszeit ein Zuruf registriert, wird die Zeilennummer welche durch den zweiten Parameter spezifiziert wird angesprungen. Die Auswertung eines registrierten Zurufs kann dann vom Anwender mit Hilfe der Systemvariable int54 erfolgen. Siehe auch hierzu die Beschreibung im Abschnitt „Systemvariabe“. Parameterübergabe: 1.Konstante/Variable 2.Zeilennummer

Sonstige Befehle

Befehle	Beschreibung
get_random_100	Es wird eine ganze Zufallszahl im Bereich von 0 und 99 erzeugt und in die angegebene Variable abgelegt. Parameterübergabe: Variable
sleep	Der Roboter geht in einen etwas energiesparenden Modus über. Das Programmablauf wird dennoch fortgesetzt und auch die Sensoren arbeiten uneingeschränkt weiter. Sobald Sie allerdings irgendeinen Bewegungsvorgang starten möchten, müssen Sie den Roboter zuvor mit dem Befehl „wake_up“ wieder aufwecken. Parameterübergabe: Keine
wake_up	Wenn der Roboter zuvor in den Energiesparmodus gesetzt wurde, wird dieser Befehl ihn wieder aufwecken und es können wieder Bewegungsvorgänge aktiviert werden. Parameterübergabe: Keine
toggle_walk_mode	Rob-230 unterstützt zwei Gangarten, die durch Aufruf dieses Befehls gewechselt werden. Voreingestellt ist die Gangart, die den Roboter am Besten geradeaus laufen lässt. Parameterübergabe: Keine
make_action_number	Übergabe eines Remote-Codes um direkt die Robotermechanik anzusteuern. Anwendungsbeispiel: Durch Empfang von Remote-Codes über den Befehl „notice_remote_time_goto“ mit anschließender Codeweitergabe an den Befehl „make_action_number, lässt sich der Roboter über die Fernbedienung steuern. Parameterübergabe: Konstante/Variable
reset	Setzt den Roboter wieder in einen definierten Anfangszustand bezüglich seiner Gelenke zurück, die auch immer nach einem Einschaltvorgang in seiner Initialisierungsphase eingenommen wird. Parameterübergabe: Keine
-	Um eine Leerzeile in Ihrem Steuerprogramm einzufügen, geben Sie lediglich einen Bindestrich als Befehl ein. Dadurch haben Sie ein einfaches Hilfsmittel einzelne Programmabschnitte optisch voneinander abzugrenzen und somit die Übersichtlichkeit deutlich zu erhöhen. Parameterübergabe: Keine

Firmware-Update durchführen

Wir sind stets bestrebt unsere Firmware zu pflegen und ggf. mit erweiterten Funktionen auszustatten. Damit Sie immer von der aktuellsten und leistungsfähigsten Firmware-Version profitieren, können Sie nachfolgend beschriebene Prozedur für ein Update durchführen.

1. Besuchen Sie unsere Website www.homerobot-service.de und vergewissern Sie sich, ob für Ihr Robotersystem eine aktuellere Firmware-Version (= höhere Versionsnummer) vorliegt. Laden Sie die log-Datei herunter und speichern diese in ein Verzeichnis Ihrer Wahl.

2. Installieren Sie das Terminalprogram „ShamCom“ auf Ihren PC, falls noch nicht geschehen, um einen reibungslosen Update-Vorgang zu gewährleisten.

3. Verbinden Sie das serielle Schnittstellenkabel einerseits mit dem ausgeschalteten Rob-230 und Ihrem PC. Falls Ihr PC nur über eine USB-Schnittstelle verfügt, schalten Sie den „USB to SERIAL Converter“ dazwischen und installieren die entsprechenden beiliegenden Treiber, falls noch nicht geschehen.

4. Starten Sie das Terminalprogram „ShamCom“ auf Ihrem PC.

5. Aktivieren Sie nun den Firmware-Update-Modus ihres Rob-230 Systems, indem Sie als erstes den Firmware-Update Aktivierungsstecker (5-poliger DIN-Stecker) in die 5-polige DIN-Buchse an der linken vorderen Schulter des Rob-230 einstecken.
   Danach schließen Sie den Ladeadapter an und stecken den Stromstecker in die vordere Strombuchse. Es sollte jetzt die rechte rote LED leuchten, was die Bereitschaft des Robotersystems für den Datenempfang signalisiert.

6. Senden Sie nun mit Ihrem Terminalprogram die zuvor gespeicherte log-Datei an das Rob-230 System. Rufen Sie dazu die Menüpunkte „Ausführen“ und „Textdatei abspielen…“ auf.
   Es öffnet sich ein Fenster, wo Sie die gespeicherte log-Datei auswählen und mit dem Button „Öffnen“ den Upload-Vorgang starten. Die rechte LED sollte dabei ausgehen und die linke LED sich im Gegenzug dafür einschalten. Dies signalisiert den zur Zeit korrekt ablaufenden Datenempfang.

7. Warten Sie bitte bis auch diese LED erlischt, was das Ende des Update-Vorganges signalisiert. Die Zeit für das Update kann mehrere Minuten in Anspruch nehmen, bitte haben Sie etwas Geduld und unterbrechen Sie in dieser Zeit auf KEINEN Fall die Stromversorgung. Sollte mindestens eine LED anfangen zu blinken, oder der Update-Vorgang länger als 5 Minuten andauern, liegt ein Fehler vor und Sie können den Vorgang abbrechen. In einem solchen Fall überprüfen Sie bitte die von Ihnen herunter geladene log-Datei und die vorgenommenen Einstellungen und wiederholen den Update-Vorgang.

Programmbeispiele

Einfacher autonomer Spaziergang mit Hinderniserkennung

10	set_sonar_distance_near	40		(Setzen einer minimalen Hindernisentfernung von 40cm)
20	-			(Leerzeile für eine bessere optische Abgrenzung des nachfolgenden Prorammabschnittes)
30	walk_forward_time_near	300	100	(300 zehntel Sekunden geradeaus bewegen , oder zur Zeile 100 springen, wenn ein Hindernis innerhalb 40cm erkannt wurde)
35	wait	30		(Pause von 3 sec einlegen)
40	goto	30		(Zurück zum Schleifenanfang)
90	-			(Leerzeile für eine bessere optische Abgrenzung des nachfolgenden Prorammabschnittes)
100	turn_right_time	40		(Rechtsdrehung für 4 sec, da Hindernis voraus!)
110	goto	30		(Zurück zum Schleifenanfang)

Auswertung der Umgebungsgeräusche mit entsprechender Reaktion

10	notice_noise_time_goto	50	100	(5 sec auf eine Geräuschpegelüberschreitung warten. Sprung zur Zeile 100, wenn Umgebungsgeräusch überschritten wurde)
20	tilt_body_right			(Korpus kurz nach rechts neigen)
30	tilt_body_left			(Korpus kurz nach links neigen)
40	wait	10		(10 zehntel Sekunden warten)
50	goto	10		(Zurück zum Anfang und Wiederholung der Prozedur)
99	-			(Leerzeile für eine bessere optische Abgrenzung des nachfolgenden Prorammabschnittes)
100	right_arm_up			(Rechten Arm heben)
110	wait	20		(20 zehntel Sekunden warten)
120	right_arm_down			(Rechten Arm senken)
130	wait	10		(10 zehntel Sekunden warten)
140	goto	10		(Zurück zum Anfang und Wiederholung der Prozedur)

Annäherung eines Objektes auswerten mit Reaktion bei Abstandunterschreitung

10	copy	100	int31	(Vergleichsvariable (=Systemvariable) mit einer Entfernung von 100cm laden, bei dessen Unterschreitung eine Aktion ausgeführt wird)
20	get_sonar_distance	int1		(Entfernungsmessung durchführen und Ergebnis (cm) in Variable int1 für die nachfolgende Auswertung sichern)
30	if_int31_grater_goto	int1	100	(Sprung zur Zeile 100 wenn Wert in int31 größer ist, als die gemessene Entfernung in int1)
40	wait	3		(3 zehntel Sekunden warten)
50	goto	20		(Zurück zum Schleifenanfang)
99	-			(Leerzeile für eine bessere optische Abgrenzung des nachfolgenden Prorammabschnittes)
100	walk_backward_time	30		(3 Sekunden lang sich rückwärts vom Objekt entfernen)
110	goto	20		(Zurück zum Schleifenanfang)

Kürzeres Programm mit ähnlicher Funktion, welches einen speziellen „walk_back“ Befehl mit Entfernungsmessung verwendet

10	set_sonar_distance_far	100		(Entfernungsüberschreitungsgrenze von 100cm setzen)
20	walk_back_time_far_goto	10000	30	(Rückwärtsbewegung wird gestoppt und zur Zeile 30 gesprungen, sobald eine Entfernung von einem vorausstehenden Objekt gemessen wird, das weiter als 100cm vom Roboter entfernt ist. Hinweis: Die Bewegung wird ggf. erst gar nicht gestartet)
30	goto	20		(Zurück zum Schleifenanfang)

Mögliche Störungen

Roboter reagiert nicht nach dem Einschaltvorgang

Fehlergrund	Lösungsvorschlag
Die Akkupacks sind nicht alle angeschlossen bzw. nicht aufgeladen.	Vergewissern Sie sich, dass beide Akkupacks, welche sich in den beiden Füßen des Rob-230 Systems befinden, aufgeladen und über die Steckverbinder angeschlossen sind.
Die Sicherung ist durchgebrannt.	Setzen Sie die beigefügte Ersatzsicherung (2A flink) in den Sicherungshalter ein. Dieser befindet sich im linken Fuß. Sie müssen dabei die Kappe des Sicherungshalter heraus drehen. Sollte die neu eingesetzte Sicherung gleich wieder durchbrennen, so kontaktieren Sie bitte den Kundendienst. Verwenden Sie auf keinen Fall eine stärkere Sicherung, denn dies könnte ernsthafte Schäden nach sich ziehen!

Kontakt

EDV-Technik-TS
Dipl. Ing. Thomas Spangemacher
Marktplatz 3
D-63065 Offenbach

www.EDV-Technik-TS.de
eMail: kontakt@edv-technik-ts.de